

The Need for a Name to MAC Address Mapping Towards Quantifying the Resource Gain

Peter Kietzmann
HAW Hamburg
peter.kietzmann@haw-hamburg.de

Cenk Gündoğan
HAW Hamburg
cenk.guendogan@haw-hamburg.de

Thomas
HAW
t.schmidt@

Oliver Hahm
INRIA
oleg@riot-os.org

Matthias Wählich
Freie Universität Berlin
m.waehlich@fu-berlin.de

ABSTRACT

In this paper, we start from two observations. First, many application scenarios that benefit from ICN involve battery driven nodes connected via shared media. Second, current link layer technologies are completely ICN agnostic, which prevents filtering of ICN packets at the device driver level. Consequently, any ICN packet, Interest as well as Data, is processed by the CPU. This sacrifices local system resources and disregards link layer support functions such as wireless retransmission. We argue for a mapping of names to MAC addresses to efficiently handle ICN packets, and explore dynamic face-based mapping schemes. We analyze the impact of this link-layer adaptation in real-world experiments and quantitatively compare different configurations. Our findings on resource consumption, and reliability on constrained devices indicate significant gains in larger networks.

CCS CONCEPTS

- **Networks** → **Layering**; *Naming and addressing*;

KEYWORDS

ICN; NDN; Internet of Things; wireless; link layer

ACM Reference Format:

Peter Kietzmann, Cenk Gündoğan, Thomas C. Schmidt, Oliver Hahm, and Matthias Wählich. 2017. The Need for a Name to MAC Address Mapping in NDN: Towards Quantifying the Resource Gain. In *Proceedings of ICN'17*. ACM, New York, NY, USA, 7 pages.

1 INTRODUCTION

Named-data networking (NDN) has the potential to improve performance of application scenarios that connect devices via lossy media such as radio. By providing (in-network) caching

on the network layer, NDN helps to on the data link layer by placing c which leads to reduced hop-counts a loss on the application layer. However, communication, there is no clear mapping of name and a MAC address. This often (multicast) frames on the *data link* mapping prevents ICN from being

Broadcasting on the data link layer distribution but also introduces two frames are not filtered by common work interface card (NIC). They are which conflicts with limited hardware CPU, memory, and energy. It is w nodes in a network can provide c of low-end hardware capabilities [1 layer technologies such as 802.11 and error handling of broadcast frames is not even supported in 802.15.4. differences on the data link layer c rent solutions address these problems either implement an NDN-specific error-resilience [9, 10, 18, 20] or, m rent device drivers to implement na NIC [17]. A dynamic mapping of to NDN faces [7, 19] is not sufficient question on how well broadcast c This surprisingly unsatisfying stat revisit the problem and solution sp

In this paper, we investigate h solutions and design options impac periments, we conduct micro bench devices to better understand the MAC address-to-face mapping. On tributing L2 packets via unicast in to significant performance gains wit resources. The objective of this p discussion starter to find a common distribution on both the network lay

The remainder of this paper is : Section 2, we discuss the problem sp For the case of dynamic face map design space in more detail in Se measurements in Section 4, and coi

2 PROBLEM SPACE AND RELATED WORK

2.1 The IoT Use Case

The Internet of Things gathers a diverse set of very heterogeneous nodes. Our focus is on low-end IoT devices, equipped with hardware resources of class 2 [8], connected via radio, and powered by battery. These devices benefit from NDN in the following way. First, the lightweight NDN network stack requires less memory compared to the current IoT stack standardized in the IETF. It is worth noting that cache sizes are independent from network stack sizes. NDN provides off the shelf name-based management and monitoring capabilities, without introducing dedicated services on top of the network layer. Second, those devices may offload data to more powerful nodes without the burden of additional protocols. This is implemented natively on the network layer, and thus simplifies application programming. Third, this in-network caching allows nodes to sleep longer, reduce data delivery latency, and increase content availability [13].

It is worth stressing that the IoT does not only gather different nodes among different operators but may also be heterogeneous within a single domain. As such, we cannot assume that all nodes provide the same set of services, neither on the application, nor on the network layer.

2.2 Current Solutions and Challenges

2.2.1 NDN-specific Link Layer. Shi *et al.* [18], Grassi *et al.* [9, 10], and Wang *et al.* [20] argue for a link protocol that is specific to NDN. Shi *et al.* [18] introduce NDNLP, which features fragmentation and reassembly as well as acknowledgement and retransmission of packets. NDNLP is located between the network layer (*e.g.*, NDN) and the virtual (*e.g.*, tunnels) or physical (*e.g.*, Ethernet) link layer. Grassi *et al.* [9, 10] present a link adaptation layer, which is tailored for vehicular networks but follows conceptually the same idea as NDNLP. Similarly in the context of improved reliability, Wang *et al.* [20] introduce an NDN broadcast protocol, which tries to minimize collision. Both approaches aim for an increased packet delivery ratio by measures below the application and network layer but still require packet processing by the CPU, independent whether a specific NDN service bound to the broadcast packet is available or not.

2.2.2 Name-based Filtering on NIC. Shi *et al.* [17] propose name-based filters on the device driver level of the network interface card. To optimize the implementation for limited on-chip memory of the NIC, names are maintained in a Bloom filter table. This approach exhibits good performance results but comes with the drawback of a layer violation. The data structure to implement filtering is specific to the ICN approach above the link layer. However, not all ICN approaches follow the same naming scheme [4, 21]. Consequently, changing the specific ICN network stack may require update of the device driver. This will slow down deployment of upcoming approaches. More importantly, this approach distributes data via layer 2 broadcast frames, and thus does not benefit from error handling on this layer.

2.2.3 Unicast Faces. Approaches to the problem of the link layer or device driver are proposed by Teubler *et al.* [19] and Baccelli *et al.* [22]. They introduce unicast faces. Basically, unicast faces map NDN names to MAC addresses to NDN faces. These are then used to filter out unwanted Interests are broadcasted, containing the MAC address of the sender. Having the MAC address of the sender, the receiver makes use of the source face. NDN packets which are not from the source face conversely include the unicast face. This approach combines both native MAC-based filtering and NDN filtering, allowing a handling/prevention on the data link layer. This is suitable for specific adaptations to the link layer and in case not all nodes within a broadcast domain provide the same network layer services, such as unicast. On the other hand, unicast traffic reduces data redundancy. A detailed analysis of the trade-off between broadcast and unicast on the system load is not present. In this paper, we argue for a more general MAC-layer mapping. There are several scenarios in which a reduced system load output is not the primary goal. Our analysis in Section 4 is a first step.

3 DESIGN SPACE BY INSIGHTS FROM EXISTING LINK LAYER

An NDN node can send Interest packets via unicast or broadcast on the link layer. In this paper, we discuss pros and cons of each approach. In Section 4, we perform a first experimental reality check on the need for link layer support.

For the sake of clarity, we focus on the design space of NDN nodes with a single interface connected to the network via shared media. Extending this design space to multiple interfaces does not change the fundamental design space. Moreover, we do not explicitly discuss the design space for First, typical lower layer IoT protocols do not support multicast. Second, link layer mapping of MAC address instead of a unicast MAC address to a name to group address mapping is not supported.

3.1 Broadcast or Unicast for Interest

Case 1: Interest Broadcast, Data Unicast. Within the broadcast domain will forward data as link layer broadcast. As long as the link layer name prefix in the forwarding table matches these (successfully received) Interests, the node will try. Consequently, as soon as a correct Interest is transmitted within the broadcast domain, the node will forward this data packet. This highest level of redundancy ensures that data is delivered. Practically, in a densely connected network, this level of redundancy without degrading data delivery, as long as the link layer content. However, this level of packet redundancy leads to excessive overhead for each node (e.g., increased energy consumption) also for the complete network (*e.g.*, increased interference). Interferences should be considered e

broadcast traffic, as there is no protective repair of errors on the link layer.

Case 2: Interest Broadcast, Data Unicast. Similar to case 1, all nodes of the broadcast domain will create a PIT entry after receiving an Interest packet. However, data packets are directed to the unicast MAC address which is associated with the corresponding outgoing face (*i.e.*, the MAC address of the next hop). As we assume a shared media, all other nodes within radio reach receive the data packet, as well, but drop it at the device driver level because of an unmatching (unicast) MAC address. Those nodes can neither cache nor forward the data on the network layer. Previously created PIT entries will thus not dissolve by receiving data but by timeouts. These PIT entries require memory, processing time and will not help to achieve redundancy. To cope with node failures, an additional mechanism is needed to keep the MAC-face assignment in sync with the MAC address of an alternative next hop.

Case 3: Interest Unicast, Data Unicast. Compared to the previous scenarios, in this case, Interest as well as data packets are sent to a unicast MAC address, using unicast faces as described in Section 2.2.3. Such an approach implements hop-by-hop forwarding on the link layer and prevents redundancy completely because any overheard packet is dropped by the network interface card. This setup requires active maintenance of MAC-to-face mapping in case of node failures.

In contrast to Case 2, updating only the unicast data face is not sufficient. Data will be forwarded based on PIT entries. The strong coupling of Interest and data flow requires that the MAC address assigned to the Interest face is in line with the data face. However, usually there is a time gap between sending Interest and forwarding corresponding data. A unicast MAC address that is valid during Interest submission might be outdated when data is forwarded. On the other hand, this case reduces radio transmissions and CPU processing to a minimum and fully incorporates MAC layer retransmission handling.

Case 4: Interest Unicast, Data Broadcast. The last case provides very limited redundancy. Data packets will be processed by the NDN stack of all nodes of the broadcast range. However, as Interest has been delivered via MAC unicast, only one node in the broadcast domain created a PIT entry. All other nodes will thus drop the data packet at network layer.

Discussion. Case 1 promises path and data redundancy but comes to the cost of excessive resources consumption which may be harmful, especially in IoT networks. Case 2 optimizes data transport via unicast but keeps forwarding redundancy and superfluity of a routing protocol. Case 3 fully optimizes resource overhead and transmission robustness which is promising for battery driven, constrained nodes. However, this approach requires a reliable routing mechanism since it minimizes path redundancy as well caching capabilities. Case 4 brings little benefit to NDN, as redundant data is not utilized.

Interest \ Data	Broadcast
	Unicast
Broadcast	12.1 % (<i>Case 1</i>)
Unicast	3.3 % (<i>Case 4</i>)

Table 1: Unsatisfied Interests with address mappings under presence of li

3.2 The Case for Link Layer

Experimental Exploration. In this experiment, we want to check back on the effect of a by counting incomplete Interest-data pairs. We select three nodes within radio reach in the FIT IoT-Lab testbed (s. Section 2.1). Each node requests 1000 content items from one of two Interests per second (without installing different MAC layer mapping). This generates side traffic on the same radio channel of 50 Bytes within random intervals.

Results. Table 1 presents unsatisfied Interests as an indicator of packet loss. Interestingly, we see that broadcasting Interest-data pairs is about one order of magnitude more efficient than Broadcasting data after Interest submission. This is due to the fact that Broadcasting data after Interest submission is subject to a timeout, which we account to an implicit retransmission. In the presence of a periodic radio channel, data is retransmitted on the MAC layer until the transfer succeeded. Data is then forwarded immediately and thus takes less time (Case 2 versus Case 4). Broadcasting (Case 1) combines these two sources of loss. We conclude that NDN can benefit from utilizing the support of MAC layer mapping.

4 EXPERIMENTAL EVALUATION

The objective of our experiments is to evaluate the effects through different MAC layer mapping in an IoT environment. Thereby, we make use of solutions and typical IoT hardware and radio transmission technologies. We focus on Case 3 (see Section 3) because CC can handle data transmissions via broadcast in a more efficient way.

4.1 Basic Testbed Setup

All experiments are conducted in the testbed to reflect common IoT properties. It consists of several hundreds of class 2 devices (Cortex-M3 MCU, 64 kB of RAM and an IEEE 802.15.4 radio (*i.e.*, Atmel radio card provides basic MAC layer mapping in hardware, such as ACK handling, CSMA/CA). For power measurement, we use a consumption monitoring tool of the testbed with a resolution time of 332 μ s and averaging.

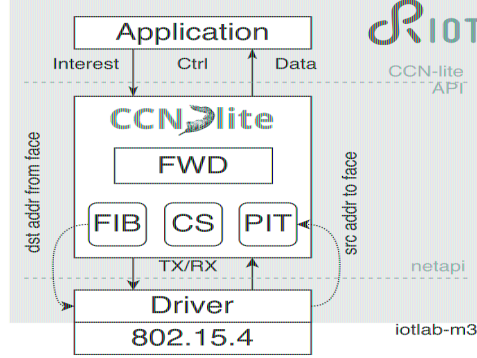


Figure 1: System environment: Integration of RIOT and CCN-lite to implement dynamic broadcast and unicast faces in NDN.

software platform is based on RIOT [6] and the CCN-lite network stack [1], which we include as a third party library in RIOT. The integration of CCN-lite into RIOT and its default components are visualized in Figure 1.

We use default configuration parameters in RIOT and CCN-lite where possible and not mentioned otherwise. In detail, we deploy RIOT release 2017.01 and CCN-lite master with latest updates from May 10, 2017. For our measurements, we configure CCN-lite with a maximum of three Interest retransmissions and 12 seconds Interest timeout. MAC configurations of the radio devices enable IEEE 802.15.4 ACK requests for unicast traffic with a maximum of four retransmissions, and CSMA with a maximum number of four retries, introducing random delays after denied channel access (see [5] for further default values).

The subsequent experiments include single-hop and multi-hop scenarios, which we describe in more detail next to the analysis of our experiments. Our results represent averages over multiple runs with the same parameter settings.

4.2 Single-hop Scenario

4.2.1 Configuration. We deploy our single-hop measurements at the *Lille* site of the FIT IoT-LAB testbed because all nodes are located in the same broadcast range. We randomly select a single consumer node and a varying number of producer nodes for different measurement runs. Each producer is equipped with a different number of unique and static content items. In all subsequent scenarios, the consumer requests existing content items randomly. We measure the number of system wakeups and the CPU load of both the CCN-lite software stack and the radio device driver.

To implement single-hop data exchange on the data link layer and the network layer, all nodes need to be in physical reachability and consumers need to have routing entries that reach the producers directly. To consider common scenarios, we analyze three basic configurations. (i) On *all nodes*, we install a common prefix route that covers all content names,

and the corresponding face refers to the producer. Note, in this case, a unicast MAC address is required for the ability of arbitrary content items via unicast. For other variations, we install dedicated FIB entries for each *producer* or *consumer*, which refer (ii) either to the producers or (iii) to the broadcast face.

Furthermore, to analyze different network sizes, we vary the number of producers and the number of content items per producer in a pre-defined scenario.

Variable network size. The number of nodes is fixed but we increase the different parameter settings. We increase the content request rate per producer, the number of content items sent by the consumer increases the number of nodes in the network.

Variable number of contents items. The number of nodes is set to 20 and the number of content items per node is increased over different measurement runs. We keep a constant content request rate at 1 request per second.

4.2.2 Results. Figure 2 shows the number of wakeups per producer for the single-hop scenario for different network sizes and a fixed number of content items (see Figures 2(a) and 2(b)), as well as for different network sizes and variable number of content items (see Figures 2(c) and 2(d)). Figures 2(a) and 2(b) show the setup where Interest requests are sent to the broadcast face. Figures 2(c) and 2(d) show the setup where Interest requests are sent to the unicast face. Figures 2(a) and 2(b) represent the setup where all nodes have routing entries for all content names. Figures 2(c) and 2(d) represent the setup where only the consumer has FIB entries, which maps faces to content names. Correspondingly, Figure 3 represents the energy usage we measured.

In terms of energy and processing overhead, it is visible that faces with unicast MAC addresses require less energy than broadcast faces. While the number of wakeups is constant for varying network sizes, it increases with the relation to the number of provided content items. This is due to the mapping of unicast mapping. Broadcast mapping requires more energy than unicast mapping. Broadcast mapping requires more energy than unicast mapping. Broadcast mapping requires more energy than unicast mapping.

To summarize, unicast faces can be used for battery driven IoT devices by keeping the processing overhead at a minimum, benefiting from built-in MAC layer traffic, such as ACK handling and other hand, it requires a maintenance of MAC addresses to faces by omitting built-in content replication and alternative data paths.

Deploying common prefix routes for all nodes reduces the overall performance, as each node in the broadcast range has to wake up during incoming Interests, leading to a higher energy consumption.

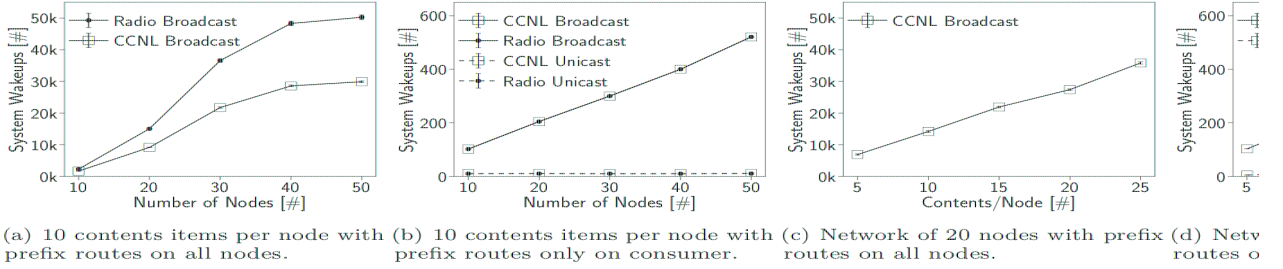


Figure 2: Number of system wakeups for varying network setups.

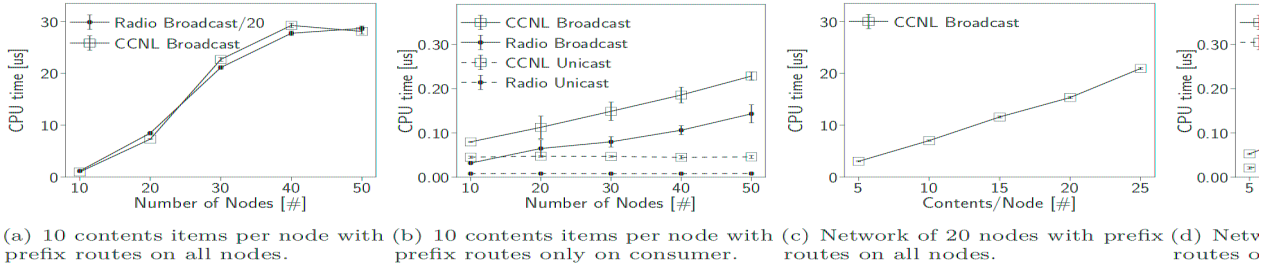


Figure 3: Absolute CPU usage for varying network setups.

packets that will not be satisfied, as well as data packets that might be received as a consequence of the forwarding mechanism. This leads to an excessive number of wakeups of all nodes in the domain as well as additional data transmissions. The overhead in this broadcast scenario is several orders of magnitudes higher than that of unicast.

To better understand the overhead introduced by broadcast Interest forwarding when all nodes store FIB entries, Figures 2(a) and 2(b) present wakeups, which are separately shown for the radio device and the CCN-lite (CCNL) network stack. In a setup consisting of a single application, single network stack, and a single network interface, both measurements should be roughly equal as the link layer forwards each broadcast packet up to the network stack and vice versa. This holds only in case of a single forwarder (see Figure 2(b)). We detect a much higher number of wakeups by the device driver when all nodes store routes. The impact on CPU times is worse by a *factor of 20*, as depicted in Figure 3(a). We assume radio channel saturation causing this increased resource consumption. To further back these observation, we also measured (on the same network scale) (i) the rate of unsatisfied Interests (0 – 50 %), (ii) radio statistics from which we compute the rate of unsuccessfully transmitted packets due to failing CSMA/CA channel access (0,39 – 0,56), and (iii) the average number of network layer retransmissions (2 – 9 %). All these observables indicate a negative impact on network utilization while broadcasting.

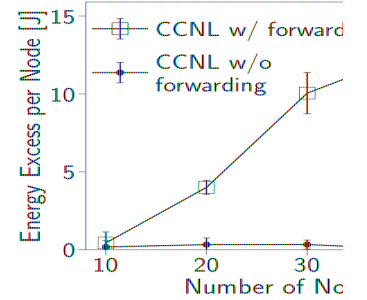


Figure 4: Average energy excess per node and without common prefix routes.

Figure 4 displays the energy per node consumed when Interests are broadcasted with and without forwarding over an Interest forwarding network. By comparing Figures 2(a) and 2(b), and Figure 4, it is evident that forwarding exceeds single-hop forwarding by several orders of magnitude, for increasing (single-hop) network sizes.

4.3 Multi-hop Scenario

4.3.1 Configuration. We conducted experiments in *Grenoble*. This site of the test

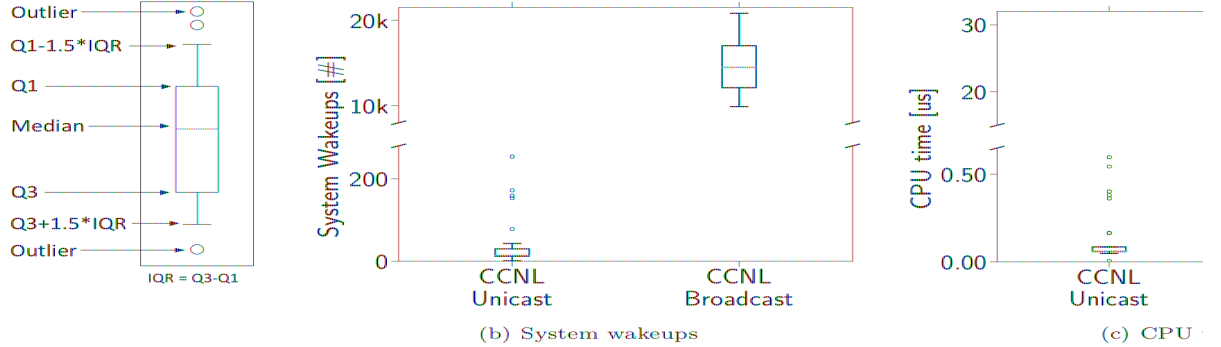


Figure 5: Network of 30 nodes w/ 20 producers and 10 contents items per producer

of nodes such that nodes do not form a single broadcast domain. However, fluctuating properties of the wireless media (*e.g.*, reflections) may lead to changing topologies from multi-hop to single-hop. To ensure a minimal multi-hop connectivity, we introduce a monitoring phase before our experiments start which is based on the mechanism proposed in [11]. During this phase, we identify a set of nodes that inter-connect over multiple hops. The resulting topology consists of 30 nodes where one node acts as consumer, 20 nodes act as producers with a distance of three hops towards the producer, and other nodes serve as intermediate nodes on the path.

Similar to the single-hop scenario, leaf nodes of the resulting topology are equipped with unique content items that are requested by a single consumer in randomized order. Referring to the single-hop experiments, we compare two mapping schemes from content names to faces at the consumer node and subsequent intermediate nodes: (i) a direct assignment of the next-hop MAC address to the corresponding face on the path to the producer, and (ii) a common prefix route where the corresponding face is mapped to a broadcast MAC address.

Even though the same set of nodes is used for (i) and (ii) we cannot guarantee that the same topology appears within the broadcast scenario, as discussed earlier. We do not consider this as a drawback but rather as an advantage, reflecting real-world properties.

4.3.2 Results. In Figure 5, we show the impact of broadcast and unicast faces in a multi-hop network in terms system wakeups and CPU times for a fixed size network and a predefined number of content items per producer. We find similar effects compared to the single-hop scenario, where resource costs for the broadcast mapping (with common prefix routes applied on all nodes) are orders of magnitudes higher than for the unicast mapping. The medians of both wakeups and CPU times correspond to our single-hop measurements but larger errors and outliers are visible. The reason for the outlier is rooted in intermediate nodes. These nodes only forward Interest and data packets on the path between producer and

consumer. In our measurements, links which were stable during the asymmetric link behavior later. The unicast approach delivered 100 % of the due redundant paths. The resource to unicast address mapping as well as layer features such retransmission of a route maintenance mechanism fresh and stable links. Analyzing this will be part of future work.

5 DISCUSSIONS AND OUTLOOK

In this paper, we discussed current state of interaction between the NDN network and the data link layer. In contrast to the current mapping IP addresses to unicast MAC addresses for arbitrary broadcast, there is no such mapping in NDN. Without sacrificing the principle of NDN, we argue that link layer broadcast should be avoided in deployment scenarios (*e.g.*, IoT), as hardware resources in terms of power and memory are scarce. We reviewed the current solutions and proposed a first set of experiments in a real-world testbed. The results of a first set of experiments in a real-world testbed show that the resource overhead and the advantage of broadcast functions (*e.g.*, retransmission handling) are significant.

The position of this paper is that the current mapping is needed and should be questioned. Second, our community should not affect the core of current link layer solutions and benefits from built-in link layer solutions. The dynamic and resource-efficient solution, the dynamic and resource-efficient solution, has been mostly ignored and deserves further research. In this paper, we contributed a first real testbed and related analysis. We hope that the community will find a common understanding of the interaction between network and data link layer.

A Note on Reproducibility

We explicitly support reproducible research [3, 16]. Our experiments have been conducted in an open testbed. The source code of our implementations (including scripts to setup the experiments, RIOT measurement apps etc.) are available on Github at <https://github.com/inetrg/ACM-ICN-2017>.

Acknowledgments

We thank our shepherd Lixia Zhang and the anonymous ICN reviewers for their valuable feedback. This work was supported in parts by the German Federal Ministry of Research and Education within the project *I3: Information-centric Networking for the Industrial Internet*.

REFERENCES

- [1] 2014. CCN Lite: Lightweight implementation of the Content Centric Networking protocol. (2014). <http://ccn-lite.net>
- [2] 2015. IoT-LAB: a very large scale open testbed. <https://www.iot-lab.info/>. (2015).
- [3] ACM. Jan., 2017. Result and Artifact Review and Badging. <http://acm.org/publications/policies/artifact-review-badging>. (Jan., 2017).
- [4] Bengt Ahlgren, Christian Dannewitz, Claudio Imbrenda, Dirk Kutscher, and Börje Ohlman. 2012. A Survey of Information-Centric Networking. *IEEE Communications Magazine* 50, 7 (July 2012), 26–36.
- [5] Atmel. [n. d.]. *Low Power 2.4 GHz Transceiver for ZigBee, IEEE 802.15.4, 6LoWPAN, RF4CE, SP100, WirelessHART, and ISM Applications*. Atmel Corporation. <http://www.atmel.com/images/doc8111.pdf>
- [6] Emmanuel Baccelli, Oliver Hahm, Mesut Günes, Matthias Wählisch, and Thomas C. Schmidt. 2013. RIOT OS: Towards an OS for the Internet of Things. In *Proc. of the 32nd IEEE INFOCOM Poster*. IEEE Press, Piscataway, NJ, USA.
- [7] Emmanuel Baccelli, Christian Mehlis, Oliver Hahm, Thomas C. Schmidt, and Matthias Wählisch. 2014. Information Centric Networking in the IoT: Experiments with NDN in the Wild. In *Proc. of 1st ACM Conf. on Information-Centric Networking (ICN-2014)*. ACM, New York, 77–86. <http://dx.doi.org/10.1145/2660129.2660144>
- [8] C. Bormann, M. Ersue, and A. Keranen. 2014. *Terminology for Constrained-Node Networks*. RFC 7228. IETF.
- [9] Giulio Grassi, Davide Pesavento, Lucas Wang, Giovanni Pau, Rama Vuyyuru, Ryuji Wakikawa, and Lixia Zhang. 2013. ACM HotMobile 2013 Poster: Vehicular Inter-networking via Named Data. *SIGMOBILE Mob. Comput* (November 2013), 23–24.
- [10] Giulio Grassi, Davide Pesavento, Rama Vuyyuru, Ryuji Wakikawa, and Lixia Zhang. 2013. Vehicular Inter-Networking via Named Data. arXiv:1310.5980. Open Archive: arXiv.
- [11] Cenk Gundogan, Thomas Schmidt, and Lixia Zhang. 2013. Publish-Subscribe Deployment on a Strained Internet of Things. *Internet of Things*. IETF.
- [12] Oliver Hahm, Cédric Adjih, Emmanuel Baccelli, and Matthias Wählisch. 2013. Experiments for Link-Layer Adaptation in the Internet of Things. In *Conf. on Information-Centric Networking (ICN)*. ACM, 195–196. <http://dx.doi.org/10.1145/2485226>
- [13] Oliver Hahm, Emmanuel Baccelli, T. Wählisch, Cédric Adjih, and Laurent Internet of Things with NDN and CCN. In *Proc. of 4th ACM Conference on Information-Centric Networking (ICN)*. ACM, New York, NY, USA.
- [14] D. Kutscher, S. Eum, K. Pentikou, T. Schmidt, and M. Wählisch. 2013. *Networks (ICN) Research Challenge*. ACM, New York, NY, USA.
- [15] George C. Polyzos and Nikos Fotios. 2013. Internet of Things using Information-Centric Networking. *IEEE Communications Magazine* 51, 10 (October 2013), 18–24.
- [16] Quirin Scheitle, Matthias Wählisch, and Georg Carle. 2017. Reproducible Research in Computing. *ACM SIGCOMM Reproducibility*. ACM, New York, NY, USA.
- [17] Junxiao Shi, Teng Liang, Hao Wu, and Lixia Zhang. 2016. NDN-NIC: Name-based Filtering. In *Proc. of ACM ICN*. ACM, New York, NY, USA.
- [18] Junxiao Shi and Beichuan Zhang. 2016. NDN, Technical Report 1. NDN, Technical Report 1.
- [19] Torsten Teubler, Mohamed Hail, and Lixia Zhang. 2016. Efficient Data Aggregation with CCNx. In *19th Open European Summer School*. Vol. 8115. Springer, Berlin Heidelberg, 1–12.
- [20] Lucas Wang, Alexander Afanasy, Rama Vuyyuru, Ryuji Wakikawa, and Lixia Zhang. 2013. Information Dissemination Using NDN. In *ACM Workshop on Emerging Networking Design - Architecture, Algorithms and Applications (NetDyN)*. ACM, New York, NY, USA, 7–12.
- [21] George Xylomenos, Christos Tsilopoulos, and Lixia Zhang. 2013. Information-Centric Networking: A Survey. *IEEE Communications Surveys and Tutorials* 16, 2 (February 2013), 102–122.